Stat 102C - Lab 2

Levon Demirdjian

April 06, 2016

In this week's lab, we will be implementing the inversion method and rejection sampling in order to draw samples from a target probability distribution.

Problem 1: Sampling via the inversion method

Let the pdf $f(x)$ of a random variable X be given by $f(x) = kx^4$ where $0 \leq x \leq 1$, and where k is the normalizing constant.

1. Find the normalizing constant of $f(x)$.

2. Find the cdf $F(t)$. Plot the pdf and cdf.

3. Use the inversion method to draw samples from $f(x)$. Start off by computing the inverse of $F(t)$.

4. Use the curve function in R to plot $f(x)$. Show the plot of $f(x)$ on the same plot as the histogram of points sampled from part c. Hint: make sure the "prob" parameter is set to true in "hist", and use the "add" parameter in the "curve" function to add the curve to the existing plot. Use ?? if you're not sure what you're doing.

Solution

1. $1 = \int_0^1 f(x)dx \Leftrightarrow \frac{1}{k} = \int_0^1 x^4 dx \Leftrightarrow k = 5$.

2. $F(t) = \int_0^t 5x^4 dx = t^5$. The plots of $f(x)$ and $F(x)$ look like:

```
my_pdf <- function(x){5*x^4}
my_cdf <- function(x){x^5}

par(mfrow=c(1,2))
curve(my_pdf, 0, 1, main = "PDF of X", ylab = "Probability")
curve(my_cdf, 0, 1, main = "CDF of X", ylab = "Cumulative probability")
```
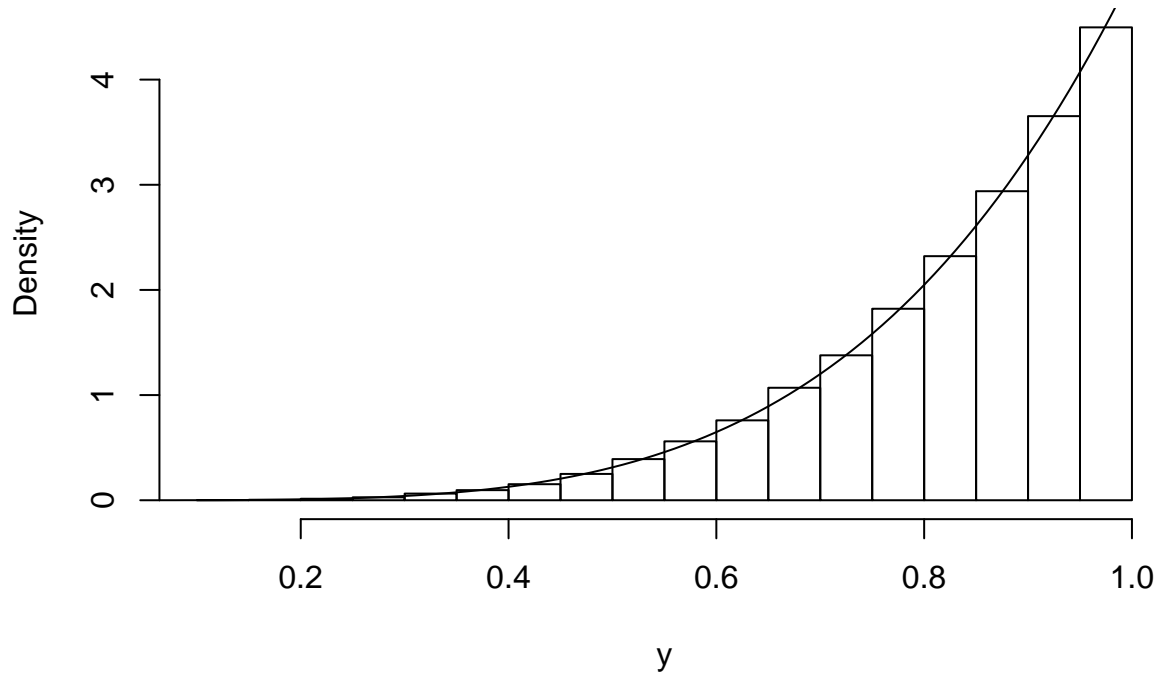
PDF of X

CDF of X
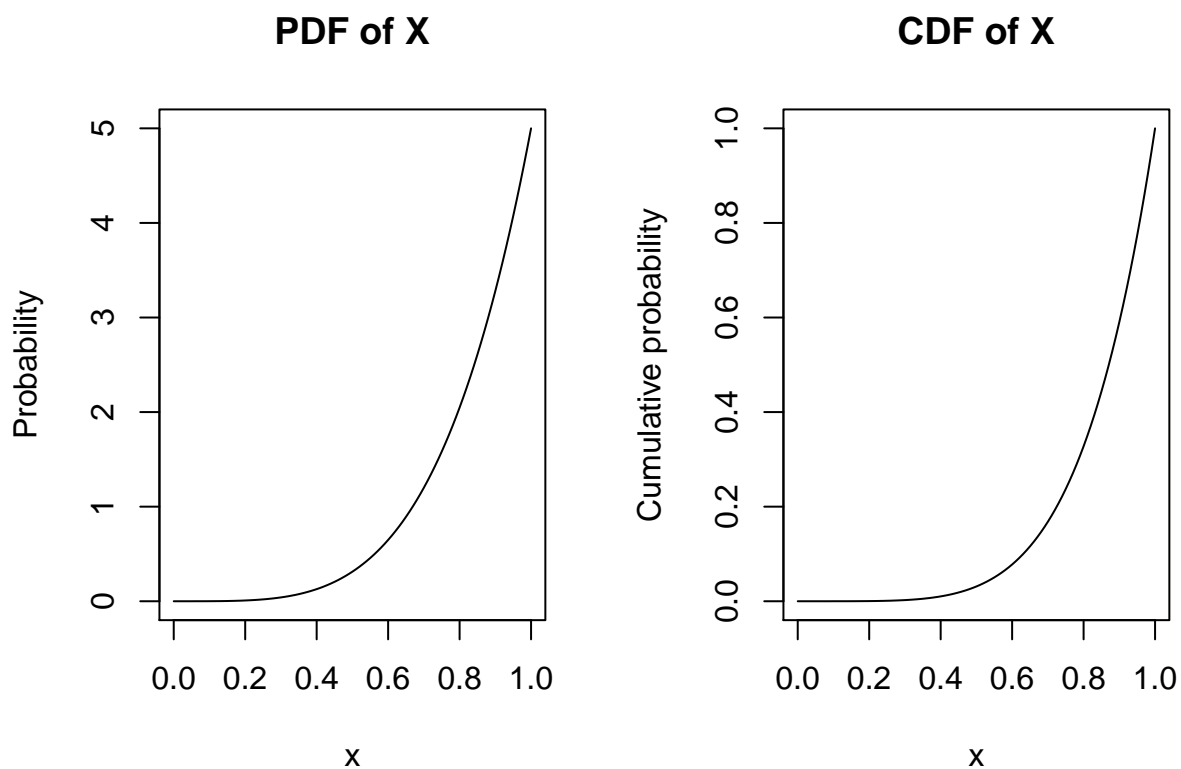
4. The inverse cdf is given by $F^{-1}(t) = t^{1/5}$. To draw samples from $f(x)$, we first draw uniform(0,1) rv's, then apply the inverse cdf to them.

```r
my_cdf_inverse <- function(x){x^(1/5)}

n <- 100000
U <- runif(n)
y <- my_cdf_inverse(U)

par(mfrow=c(1,1))
hist(y, prob = TRUE, main = "Sampled values with true pdf drawn on top")
curve(my_pdf, 0, 1, add = TRUE)
```

Sampled values with true pdf drawn on top

Problem 2: Rejection sampling

Suppose we're interested in sampling from the beta(2,5) distribution but we don't want to use R's built in function. We proceed using rejection sampling. Recall that the pdf of a beta(2,5) random variable is given by

$$f(x) = 30x(1-x)^4 \quad 0 \le x \le 1.$$

For rejection sampling, we must first define an enveloping distribution, $g(x)$, such that $f(x)$ will be completely covered by a scaled version of $g(x)$, i.e. we must find a function $g(x)$ and a constant M such that $f(x) \le Mg(x) \forall x \in [0,1]$. Can we use the Unif(0,1) distribution as the enveloping distribution $g(x)$? Sure, as long as we choose M sufficiently large to cover the beta pdf. Here is a plot of our density with $4 * g(x)$ on top:

```r
my_beta <- function(x){
  30 * x * (1 - x)^4
}
curve(my_beta, 0, 1, ylim = c(0, 5))  ## How does M=5 do?
abline(h = 4)
```
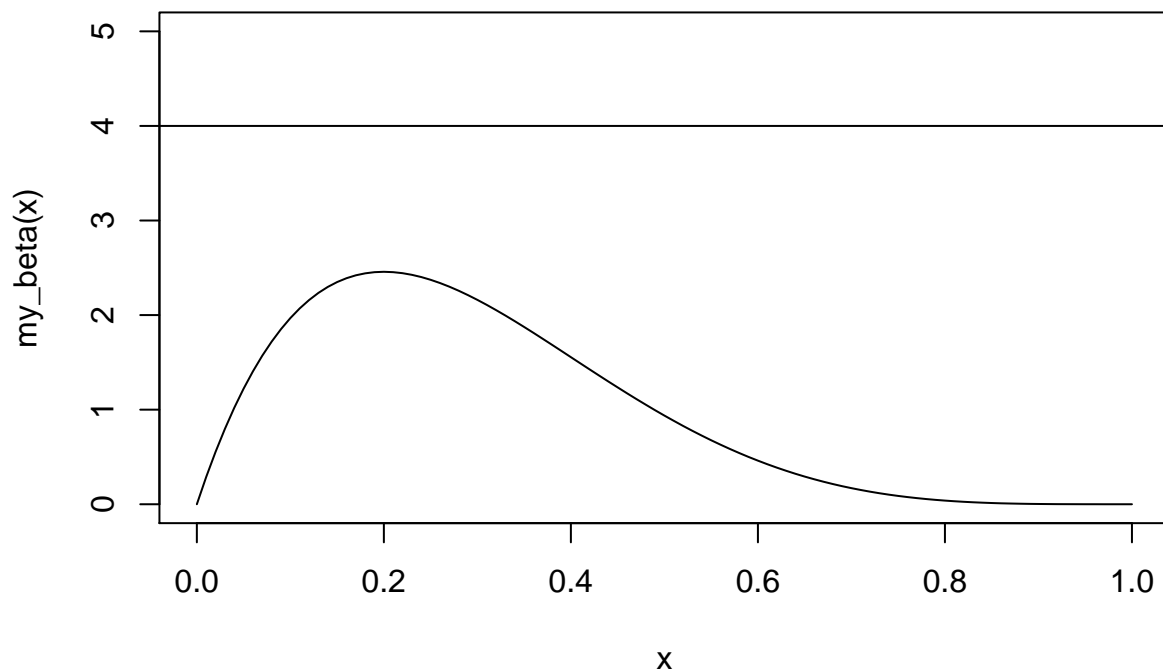
It's clear that the choice $M = 4$ satisfies our requirements. But we can do better. In fact, we can find the smallest M such that $f(x) \leq Mg(x) \forall x \in [0, 1]$. For this particular example, this is clear from the picture - we need to find a height such that the height of the rectangle just touches the peak of the density $f(x)$. Simply let $M = max_{x \in [0,1]} f(x)$ (you can calculate this value by differentiating $f(x)$ and setting it equal to 0. I'm excluding this value from these notes to make sure you do the work yourself!). Let's look at a plot of our density with $M * g(x)$ on top, where M is the optimal value.
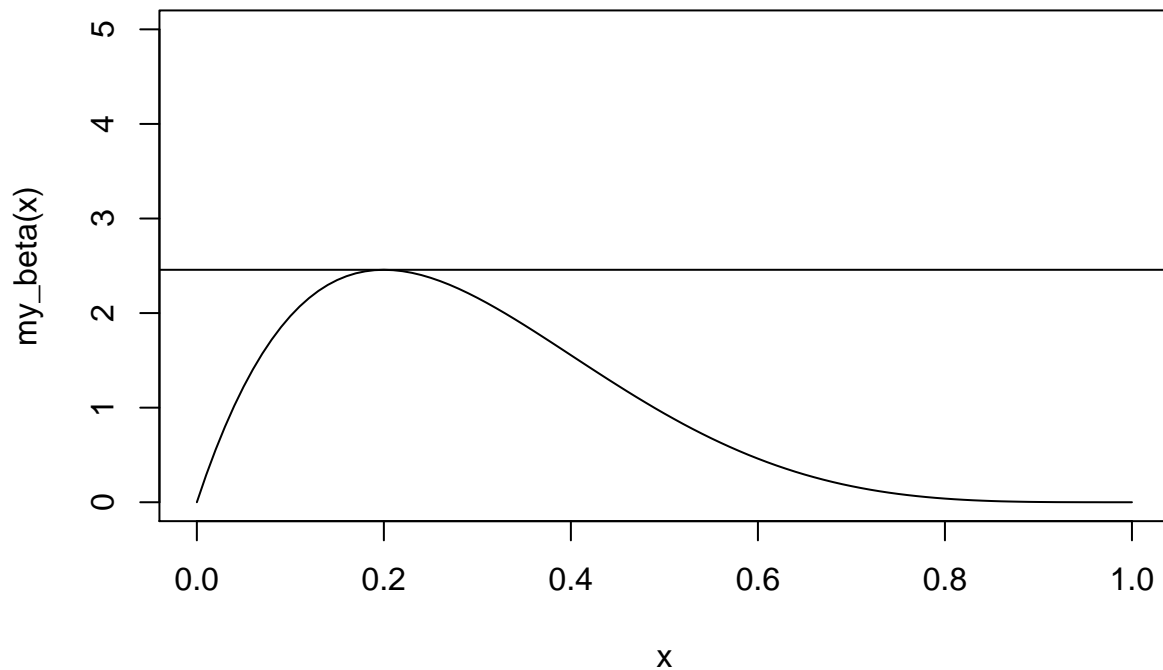
Note that both of the above schemes (using different M's) will work here, though the better the choice of M, the larger the acceptance rate (recall that the acceptance rate is $\frac{1}{M}$ so that the smaller M, the larger the acceptance rate).

We can summarize the rejection method for this example as follows:

1. Draw $X_1, ..., X_n \overset{iid}{\sim} g(x)$, where $g(x)$ is the uniform(0,1) distribution.

2. For each X_i, compute the acceptance probability $p_i = \frac{f(X_i)}{Mg(X_i)}$.

3. For each $i = 1, ..., n$, flip a coin with heads probability of p_i. If the coin is heads, accept X_i, else reject X_i and move on to the next X_{i+1}.

The following code implements this strategy for the optimal choice of M.

```
## Compute acceptance_prob
acceptance_prob <- function(x){
  (30 * x * (1 - x)^4) / M
}

n <- 10000
U <- runif(n)             ## Drawing from g(x)
y <- acceptance_prob(U)    ## Compute acceptance probability
accepted <- rbinom(n, 1, y) ## Sample from acceptance probabilities

## Plot results
```

```r
hist(U[accepted == 1], prob = TRUE, main = "Histogram of samples from f(x)", xlab = "x")
curve(my_beta, 0, 1, ylim = c(0, 5), add = TRUE)
```

Histogram of samples from f(x)
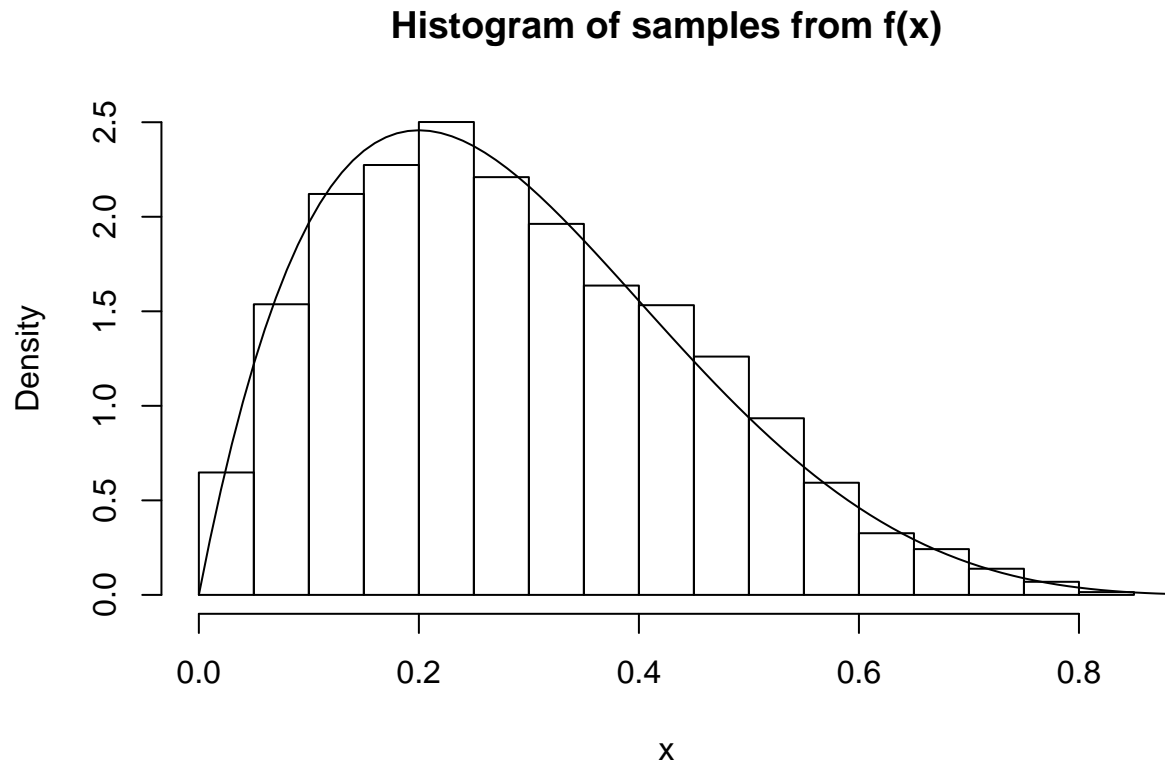
You should also calculate the rate at which we accepted draws from $g(x)$; if you did this correctly, your result should be equal to $1/M$.